UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/815,897	03/31/2004	Xuebin Yao	P19003	9318

46915 7590 02/21/2008
KONRAD RAYNES & VICTOR, LLP.
ATTN: INT77
315 SOUTH BEVERLY DRIVE, SUITE 210
BEVERLY HILLS, CA 90212

EXAMINER	
ZHANG, SHIRLEY X	
ART UNIT	**PAPER NUMBER**
2144	

MAIL DATE	DELIVERY MODE
02/21/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/815,897	YAO ET AL.	
	Examiner	Art Unit	
	SHIRLEY X. ZHANG	4121	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *31 March 2004*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-30* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-30* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *March 31, 2004* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *August 08, 2006*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

This non-final office action is responsive to the U.S. patent application no. 10/815,897 file on March 31, 2004.

Claims 1-30 are pending;

Claims 1-30 are rejected;

Information Disclosure Statement

1. The information disclosure statement (IDS) submitted on August 8, 2006 was filed after the mailing date of the application on March 31, 2004. The submission is in compliance with the provisions of 37 CFR 1.97. Accordingly, the information disclosure statement is being considered by the examiner.

Claim Rejections - 35 USC § 101

35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

2. **Claims 22-30** are rejected under 35 U.S.C. 101 because the claimed invention raises the question of whether the invention is directed to non-statutory subject matter.

Claim 22 recited "an article of manufacture comprising a storage medium". According to the specification, page 13, lines 2-3, the storage medium may comprise any information bearing medium known in the art including a transmission medium, which the examiner interprets as a signal per se, therefore is non-statutory subject matter.

Claims 23-30 are dependent on claim 22 but do not further limit the storage medium to any statutory subject matter, therefore inherit the 101 issue of claim 22.

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

3. **Claims 1-4, 6, 8-13, 15, 17-25, 27, 29-30** are rejected under 35 U.S.C. 102(e) as being anticipated by U.S. patent application publication no. 2003/0158906 to **Hayes**.

Regarding claim 1, Hayes discloses a method, comprising: requesting, by a network storage driver (Fig.10 and [0056] disclose a network application 172, which is equivalent to the network storage driver recited in the claim), a connection from an offload application ([0066] discloses that to establish a connection, in FIG. 7, network application 172 sends a request to a host resident offload task interface function 162. Here the host resident functions 162, 166 and 167 together is equivalent to the offload application recited in the claim because [0066] discloses that the AP and host resident TCP and +Application protocol processing functions 166, 158, 167, 159 are able to offload the network and application protocols that network application 172 uses), wherein the offload application interfaces with a first network stack implemented in an operating system (Fig. 10 discloses a first network stack that includes the components 118, 116 and 114 in a host operating system) and a second network stack implemented in a hardware device (Fig. 10 discloses a second protocol stack that includes the components 159, 156

and 170; [0056] further discloses that the host resident offload task interface function 162 communicates with host resident processing functions 164, 166 and 167 on one end and an AP resident offload task interface function 154 that controls the second network stack on the hardware on the other end);

receiving the connection from the offload application, wherein the received connection is an offloaded connection and is reserved for the network storage driver ([0066] discloses that in order to retrieve data from the network storage system, the network application, i.e. the network storage driver, must establish a connection, which is reserved for the network application; the last sentence of [0067] discloses that once the connection is established, a host resident task interface function 162 notifies a network application 172 of the connection); and

communicating data over the offloaded connection through the hardware device ([0068] discloses that after a connection has been established, network application 172 calls a host resident offload task interface function162 requesting that data be sent to network attached storage 16).

Regarding claim 2, Hayes discloses the method of claim 1, wherein communicating the data over the offloaded connection further comprises: sending the data directly from the network storage driver to a hardware driver for the hardware device ([0068] discloses that to request that data be send to network attached storage, the network application calls a host resident offload task interface function, which then calls an auxiliary processor (AP) resident offload task interface with a service request so that the request can be processed by the protocol stack on the AP, bypassing the host protocol stack), wherein the network storage driver uses the second network stack implemented in

the hardware device to communicate with a storage area network ([0038] discloses that the auxiliary processor offloads the reception of iSCSI data over the TCP/IP network protocol, performing all necessary TCP/IP functions that occur during the normal course of a TCP/IP receive operation and all necessary iSCSI data movement functions).

Regarding claim 3, Hayes discloses the method of claim 1, further comprising: releasing the offloaded connection to the offload application, wherein the offloaded connection is no longer reserved for the network storage driver ([0071] discloses that if the given TCP connection is to be reused and the error recovery has been completed, the TCP connection state can again be transferred from the host resident TCP protocol offload function 166 to the AP resident TCP protocol offload processing function 158, which implies that if the connection is not reused or the error recovery failed, the connection will be released).

Regarding claim 4, Hayes discloses the method of claim 1 further comprising:

receiving the request for the connection at the offload application ([0066] further discloses that to establish a connection, in FIG. 7, network application 172 sends a request to a host resident offload task interface function 162 to open a TCP connection, where the host resident offload task interface function is equivalent to the offload application recited in the claim);

generating, by the offload application, the offloaded connection ([0067] discloses that to establish the connection, a host resident offload task interface function 162 calls a host resident TCP protocol offload processing function with a protocol service request),

reserving, by the offload application, the offloaded connection for the network storage driver ([0057] discloses that protocol state information is passed between host

and the auxiliary processor, where [0103] further discloses that protocol state information is the computer data necessary to maintain a network connection by a protocol stack, i.e. it is the resources that must be reserved by the host resident TCP protocol offload processing function) and

sending the offloaded connection to the network storage driver ([0067] discloses that once a connection is established, a host resident task interface function 162 notifies a network application 172 of the connection).

Regarding claim 6, Hayes discloses the method of claim 1, wherein the network storage driver implements an Internet Small Computer Systems Interface protocol for communicating with a target storage device through the hardware device ([0039] discloses that in a preferred embodiment of the invention, the auxiliary processor offloads the transmission of "iSCSI" data over the TCP/IP network protocol, performing all necessary TCP/IP functions that occur during the normal course of a TCP/IP transmit operation and all necessary iSCSI data movement functions).

Regarding claim 8, Hayes discloses the method of claim 1, wherein the first network stack and the second network stack comprise an Internet address family and a Transmission Control protocol implemented over an Internet Protocol network layer (Fig. 10), wherein the offload application can offload a network communication request to the second network stack in preference to the first network stack, and wherein a single stack behavior is maintained by the first and second network stacks to applications and network management utilities ([0069] discloses that the host resident offload task interface function 162 recognizes that this task is most efficiently accomplished by offloading it to

an auxiliary processor 152, and calls an AP resident offload task interface function 154 with a protocol service request).

Regarding claim 9, Hayes discloses the method of claim 1, wherein the hardware device is a Transmission Control Protocol offload engine adapter ([0038] discloses that the auxiliary processor offloads the reception of iSCSI date over the TCP/IP network protocol and performs all necessary TCP/IP functions), and wherein a network communication request for communicating the data is processed faster in the second network stack in comparison to the first network stack ([0069] discloses that the host resident offload task interface function 162 recognizes that this task is most efficiently accomplished by offloading it to an auxiliary processor, which implies that it is processed faster in the second protocol stack in the auxiliary processor).

Claims 10-13, 15 and 17-18 list substantially the same elements of **claims 1-4, 6, and 8-9** but in system form rather than method form. Therefore, the supporting rationale of the rejection to claims 1-4, 6, and 8-9 applies equally as well to claims 10-13, 15 and 17-18.

Furthermore, regarding claim 10's limitation of a system comprising a processor and program logic including code that is capable of causing the processor to be operable, Hayes discloses in [0006] that the invention provides methods and apparatus for delivering selective offloading of protocol processing from a host computer to an offloading auxiliary processor and Fig. 7 discloses that the host computer comprises a CPU and memory. It is well known in the art that a host computer as disclosed by Hayes inherently comprises program logic including code that is capable of causing the processor to be operable.

Claims 22-25, 27 and 29-30 list substantially the same elements of **claims 1-4, 6, and 8-9** but in the article of manufacture form rather than method form. Therefore, the supporting rationale of the rejection to claims 1-4, 6, and 8-9 applies equally as well to claims 22-25, 27, and 29-30.

Furthermore, regarding claim 22's limitation of an article of manufacture comprising a storage medium having stored therein instructions capable of being executed by a machine, Hayes discloses in [0006] that the invention provides methods and apparatus for delivering selective offloading of protocol processing from a host computer to an offloading auxiliary processor; Hayes further discloses in Fig. 7 that the host computer comprises a memory which inherently stores instructions capable of being executed by the CPU.

Regarding claim 19, Hayes discloses a system, comprising:

a computational platform (Fig. 4 discloses client computer 12 as a computational platform);

a storage controller implemented in the computational platform (Fig. 4 discloses the network attached storage device 16);

a processor coupled to the computational platform (Fig. 4 discloses a processor 28 on the client computer 12);

an offload adapter coupled to the computational platform (Fig. 8 discloses a network interface card that includes an auxiliary processor to offload protocol processing); and

program logic including code that is capable of causing the processor to be operable to:

request, by a network storage driver (Fig.10 and [0056] disclose a network application 172, which is equivalent to the network storage driver recited in the claim), a connection from an offload application ([0066] discloses that to establish a connection, in FIG. 7, network application 172 sends a request to a host resident offload task interface function 162. Here the host resident functions 162, 166 and 167 together is equivalent to the offload application recited in the claim because [0066] discloses that the AP and host resident TCP and +Application protocol processing functions 166, 158, 167, 159 are able to offload the network and application protocols that network application 172 uses), wherein the offload application interfaces with a first network stack implemented in an operating system (Fig. 10 discloses a first network stack that includes the components 118, 116 and 114 in a host operating system) and a second network stack implemented in the offload adapter (Fig. 10 discloses a second protocol stack that includes the components 159, 156 and 170; [0056] further discloses that the host resident offload task interface function 162 communicates with host resident processing functions 164, 166 and 167 on one end and an AP resident offload task interface function 154 that controls the second network stack on the hardware on the other end);

receive the connection from the offload application, wherein the received connection is an offloaded connection and is reserved for the network storage driver ([0066] discloses that in order to retrieve data from the network storage system, the network application, i.e. the network storage driver, must establish a connection, which is reserved for the network application; the last sentence of [0067] discloses that once the connection is established, a host resident task interface function 162 notifies a network application 172 of the connection); and

communicate data over the offloaded connection through the offload adapter ([0068] discloses that after a connection has been established, network application 172 calls a host resident offload task interface function162 requesting that data be sent to network attached storage 16).

Regarding claim 20, Hayes discloses the system of claim 19, wherein the program logic is further capable of causing the processor to be operable to: release the offloaded connection to the offload application, wherein the offloaded connection is no longer reserved for the network storage driver ([0071] discloses that if the given TCP connection is to be reused and the error recovery has been completed, the TCP connection state can again be transferred from the host resident TCP protocol offload function 166 to the AP resident TCP protocol offload processing function 158, which implies that if the connection is not reused or the error recovery failed, the connection will be released).

Regarding claim 21, Hayes discloses the system of claim 19, wherein the program logic is further capable of causing the processor to be operable to:

receive the request for the connection at the offload application ([0066] further discloses that to establish a connection, in FIG. 7, network application 172 sends a request to a host resident offload task interface function 162 to open a TCP connection, where the host resident offload task interface function is equivalent to the offload application recited in the claim);

generate, by the offload application, the offloaded connection ([0067] discloses that to establish the connection, a host resident offload task interface function 162 calls a host resident TCP protocol offload processing function with a protocol service request);

reserve, by the offload application, the offloaded connection for the network storage driver ([0057] discloses that protocol state information is passed between host and the auxiliary processor, where [0103] further discloses that protocol state information is the computer data necessary to maintain a network connection by a protocol stack, i.e. it is the resources that must be reserved by the host resident TCP protocol offload processing function); and

send the offloaded connection to the network storage driver ([0067] discloses that once a connection is established, a host resident task interface function 162 notifies a network application 172 of the connection).

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

The factual inquiries set forth in *Graham* **v.** *John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the

various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

4. **Claims 5, 7, 14, 16, 26 and 28** are rejected under 35 U.S.C. 103(a) as being unpatentable over Hayes in view of the white paper "Introduction to TCP/IP Offload Engine (TOE)" authored by Yeh et al. and published in 2002, hereinafter "**Yeh**".

Regarding claims 5, Hayes discloses the method of claim 1, respectively, wherein the connection is a Transmission Control Protocol/Internet Protocol connection including state information describing the connection ([0066] discloses that a network application sends a request to open a TCP connection; [0067] further discloses that associated with each connection is state information describing the connection) sent from the offload application to the network storage driver ([0067] further disclose that once a connection is established, a host resident task interface function, i.e., the offload application, notifies a network application, i.e., the network storage driver) and wherein the state information includes a port address that is reserved for the network storage driver ([0106] discloses that a TCP connection is identified by the IP source address, destination address, source port and destination port).

Hayes does not specifically disclose a file descriptor for the TCP connection.

However, Yeh discloses in section "Performance with TCP offload" on page 4 that TOE usually interface to the system above the transport layer with a socket interface

in a sockets-based system, which implies that a socket handle, which is equivalent to a file descriptor, is created for each TCP connection.

It would have been obvious for one of ordinary skill to integrate Yeh's teaching of implementing a socket interface into Hayes so that each TCP connection in Hayes's invention includes a socket handle, or a file descriptor. One would have been motivated to combine Hayes and Yeh as such by the fact that both Hayes and Yeh taught about techniques of offloading TCP/IP stack onto hardware to expedite data processing for certain application layer protocols such as iSCSI, and socket interface is such a well-known technology in the art of TCP/IP networking that the combination would have yielded predictable results with reasonable expectation of success.

Regarding claims 7, Hayes discloses the method of claim 1, the system of claim 10 and the article of manufacture of claim 22, respectively. Hayes does not disclose that the first network stack and the second network stack do not implement an Internet Small Computer Systems Interface protocol.

However, Yeh discloses in section "Introduction" on page 1 the approach of offloading TCP/IP protocol stack to the hardware while leaving application layer protocols such as iSCSI in the software on the host, as further disclosed in the section "Applications" on page 4.

It would have been obvious for one of ordinary skill in the art to combine Hayes and Yeh such that the first network stack and the second network stack do not implement an Internet Small Computer Systems Interface protocol. One would have been motivated to combine Hayes and Yeh so that the protocol stack on the hardware will be application-

neutral and therefore has the flexibility of supporting various types of application layer protocols in the software on a need basis.

Claims 14 and 26 list all the same elements of **claim 5**, but in system or article of manufacture form rather than method form. Therefore, the supporting rationale of the rejection to **claim 5** applies equally as well to **claims 14 and 26.**

Claims 16 and 28 list all the same elements of **claim 7**, but in system or article of manufacture form rather than method form. Therefore, the supporting rationale of the rejection to **claim 7** applies equally as well to **claims 16 and 28.**

Conclusion

5. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

US 6427173 B, BLIGHTMAN, S E J et al., Communication processing device for host computer, allows processor or protocol stack of host computer to perform transport and network layer processing when received packet satisfies or not satisfies fast path criteria, respectively;

US 6965911 B1, Coffman; Jerrie L. et al., Efficiently exporting local device access onto a system area network using a direct-call interface;

US 7007103 B2, Pinkerton; James et al., Method to offload a network stack;

US 20050055456 A1, Chalupsky, David L. et al., Method, system, and program for managing a speed at which data is transmitted between network adaptors;

US 20050141434 A1, Cornett, Linden, Method, system, and program for managing buffers;

US 20050187939 A1, Krithivas, Ramamurthy, Method, system, and program for managing data read operations;

US 20040267967 A1, Sarangam, Parthasarathy et al., Method, system, and program for managing requests to a network adaptor;

US 7124205 B2, Craft; Peter K. et al., Network interface device that fast-path processes solicited session layer read commands;

US 6427171 B, BOUCHER, L B et al., Network message reception program for communication application, has instructions to transfer control of TCP connection from protocol processing stack to intelligent network interface card;

Any inquiry concerning this communication or earlier communications from the examiner should be directed to SHIRLEY X. ZHANG whose telephone number is (571)270-5012. The examiner can normally be reached on Monday through Friday 7:30am - 5:00pm EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, William Vaughn can be reached on (571) 272-3922. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/S. X. Z./
Examiner, Art Unit 4121

WILLIAM VAUGHN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100